Comparison between Standard and Adaptive MCMC via their diffusion limits

G K Basak *& Arunangshu Biswas †

Abstract

Adaptive Markov Chain Monte Carlo (AMCMC) are a class of algorithms that have been recently proposed in MCMC literature. The main difference between MCMC and AMCMC is that, the tuning parameter, which determines how fast the simulation converges to the desired distribution $\psi(\cdot)$, is a function of the previous sample paths. This destroys the Markovian character of the chain. However it can be shown that, under some conditions, the adaptive chain converges to the target distribution ψ. In this paper we use a diffusion approximation technique on a discrete time AMCMC. The resulting diffusion, which is a two-dimensional degenerate one, gives some idea of the dynamics of the chain.The diffusive limits of a Standard MCMC is a one dimensional Ornstein-Uhlenbeck (OU) process. Using extensive simulations, we compare between Standard and AMCMC, in both discrete and continuous time settings, for various target distributions (both heavy and light tailed, symmetric and asymmetric).

Keywords and phrases: MCMC, Adaptive MCMC, Diffusion approximation, tuning parameter, SDE.

AMS Subject classification: 60J22, 65C05, 65C30, 65C40

1 Introduction

MCMC is a class of widely used simulation algorithms used to generate sample from a distribution $\psi(\cdot)$ known upto only a normalising constant. Such instances are numerous in the Bayesian literature (see [4]). One member of this class is the Metropolis-Hastings (MH) algorithm which has gained widespread reputation since the work of

*Stat-Math Unit, Indian Statistical Institute, Kolkata

†Dept. of Statistics, Presidency University, Kolkata

Metropolis and Hastings ([10, 14]). Given ψ upto a constant, the method involves choosing a proposal distribution $q(x, y)$ from which samples are generated which is accepted using a MH acceptance probability given by $\min\{1, \frac{\psi(y)q(x,y)}{\psi(x)q(y,x)}\}$. Under standard assumptions, the sample thus generated is going to converge to the distribution $\psi(\cdot)$ (for more details, see [22]).

Since the algorithm involves a choice of $q(x, y)$ (which depends on a parameter σ) the time taken to converge to stationarity might be slow for bad choice of $q(x, y)$ or σ. Suggestions regarding the correct choice of σ have been previously given in literature (see [7], [6]). Another way of addressing this problem is by the use of Adaptive MCMC (AMCMC). Adaptive MCMC was first proposed by Haario et al [9] and since then there has been extensive research in this field (see [1, 2, 19, 17] to name a few). The idea behind AMCMC is to make the scaling parameter a function of the previously generated samples and acceptance ratio (the ratio of the number of samples accepted to the number of samples generated), such that if the acceptance ratio is good then the scaling parameter can be increased. On the other hand if the acceptance ratio is not good then the scaling parameter should be decreased a little. Note that, it is not required that the sequence of scaling parameters converge to some quantity. Ideally this scheme is similar to the stochastic approximation framework (see [1, 15]). However since the future value of the chain depends on the infinite past it is no longer a homogeneous Markov chain. The chain is called ergodic if the limiting distribution of the chain is the same as the target distribution. Sufficient conditions for ergodicity is given by (i) the Diminishing Adaptation condition, which states that the the difference between successive transition kernels, defined using total variation norm, diminishes to 0, and (ii) Simultaneous Ergodicity condition , which says that the time to reach the invariant distribution is uniformly bounded over the tuning parameter. For more details see [17]. However, verifying the above conditions for a given AMCMC might prove to be difficult.

In this paper we take the approach of embedding the discrete time chain in a continuous time process so that the limiting results of the resulting process can be studied. One advantage of this embedding is that the continuous time process can be studied via many discrete approximations (say Euler or Milstein,etc., which are asymptotically equivalent), whereas the discrete time chain can only be studied in the form it is specified.After embedding the discrete time AMCMC into a continuous time SDE. After embedding the discrete time AMCMC into a continuous time SDE we simulate it using the Euler approximation scheme. Also, for comparison purposes,

the standard (also called non-adaptive) MCMC is embedded in the same fashion and simulated using the same approximation scheme.

In Section 2 we define the discrete time Adaptive chain and its diffusion approximation. In Section 3 we derive the main theorem of this paper, where we obtain the limiting SDE. In Section 4 we compare the standard and the adaptive MCMC by simulating the Euler discretization of the SDE derived in Section 3. Finally we end with some concluding remarks in Section 5. Long derivation related to Theorem 1 is given in Appendix.

2 Definition of the Adaptive MCMC model and its Diffusion Approximation

First, let us define the algorithm which was proposed by Peter Green (personal communication):

1. Select arbitrary $\{X_0, \theta_0, \xi_0\} \in \mathbf{R} \times [0, \infty) \times \{0, 1\}$ where $\mathbf{R}$ is the state space. Set $n = 1$.

2. Propose a new move say Y where
   $Y \sim N(X_{n-1}, \theta_{n-1})$

3. Accept the new point with probability $\alpha(X_{n-1}, Y) = \min\{1, \frac{\psi(Y)}{\psi(X_{n-1})}\}$
   If the point is accepted set $X_n = Y,\ \xi_n = 1$; else $X_n = X_{n-1},\ \xi_n = 0$

4. $\theta_n = \theta_{n-1} e^{\frac{1}{\sqrt{n}}(\xi_n - p)} \quad p > 0 \quad \Leftrightarrow \log(\theta_n) = \log(\theta_{n-1}) + \frac{1}{\sqrt{n}}(\xi_n - p) \quad p > 0$

5. $n \leftarrow n + 1$

6. Goto step 2.

The above algorithm is equivalent to the following:

1. Select arbitrary $\{X_0, \theta_0, \xi_0\} \in \mathbf{R} \times [0, \infty) \times \{0, 1\}$, where $\mathbf{R}$ is the state space. Set $n = 1$.

2. Given $X_{n-1}, \theta_{n-1}, \epsilon_{n-1}$ generate

$$\xi_n \sim Ber\Big(\min\Big(1, \frac{\psi(X_{n-1} + \theta_{n-1}\epsilon_{n-1})}{\psi(X_{n-1})}\Big)\Big)$$

and then

$$X_n = X_{n-1} + \theta_{n-1}\xi_n\epsilon_{n-1}$$

where $\epsilon_{n-1} \sim N(0,1)$,

3. $\theta_n = \theta_{n-1}e^{\frac{1}{\sqrt{n}}(\xi_n - p)} \quad, p > 0, \quad \Leftrightarrow \log(\theta_n) = \log(\theta_{n-1}) + \frac{1}{\sqrt{n}}(\xi_n - p) \quad, p > 0.$

4. $n \leftarrow n+1$

5. Goto step 2.

Let us describe the algorithm. θ_n is the proposal scaling (tuning) parameter which is adaptively tuned depending whether the previous sample was accepted. If the sample was accepted then the proposal variance will increase allowing the chain to explore more regions in the state space. If the past sample was rejected then the variance will be small making the move a more conservative. Here p is a benchmark; for Normal target density this can be taken as 0.238, see[7]. This algorithm is somewhat similar in principle to the Stochastic approximation procedure, see [15].

Remark **1** *We check for the Diminishing Adaptation condition given in [17]. We see that* $|\log(\theta_{n+1}) - \log(\theta_n)| \sim \frac{1}{\sqrt{n}} \to 0$*. So the diminishing adaptation condition is satisfied.*

We now proceed to embed the discrete time AMCMC into a continuous time SDE. For doing so we partition the half line $[0,\infty)$ into subintervals of length $\frac{1}{n}$. We apply the discrete time AMCMC at each points in the sub interval and then interpolate it continuously. We also impose a diminishing adaptation type condition on X_n as well so that continuous embedding is posssible. The *nth* approximation $X_n(\cdot)$ to continuous time process $X(\cdot)$ looks like:

$$\begin{aligned}
X_n(0) &= x_0 \in \mathbf{R}; \\
X_n\Big(\frac{i+1}{n}\Big) &= X_n\Big(\frac{i}{n}\Big) + \frac{1}{\sqrt{n}}\theta_n\Big(\frac{i}{n}\Big)\xi_n\Big(\frac{i+1}{n}\Big)\epsilon_n\Big(\frac{i+1}{n}\Big), \quad \text{i=0, 1, \dots,} \\
X_n(t) &= X_n\Big(\frac{i}{n}\Big), \quad \text{if } \tfrac{i}{n} \le t < \tfrac{i+1}{n} \text{ for some integer } i. \qquad (2.1)
\end{aligned}$$

Here, $\xi_n(\frac{i+1}{n})$ conditionally follows the Bernoulli distribution given by:

$$P\left(\xi_n(\frac{i+1}{n}) = 1|X_n(\frac{i}{n}), \theta_n\Big(\frac{i}{n}\Big), \epsilon_n\Big(\frac{i+1}{n}\Big)\right) = \min\Big\{\frac{\psi(X_n\big(\frac{i}{n}\big) + \frac{1}{\sqrt{n}}\theta_n\big(\frac{i}{n}\big)\epsilon_n\big(\frac{i+1}{n}\big)\big)}{\psi\Big(X_n\big(\frac{i}{n}\big)\Big)}, 1\Big\}.$$

and $\{\epsilon_n(\frac{i}{n}), n \geq 1, i \geq 1 \}$ are all independent $N(0,1)$ random variables.
The nth approximation to the tuning parameter $\theta(\cdot)$ is defined as :

$$\begin{aligned}
\theta_n(0) &= \theta_0 \in \mathbf{R}^+ \\
\theta_n\left(\frac{i+1}{n}\right) &= \theta_n\left(\frac{i}{n}\right) e^{\frac{1}{\sqrt{n}}(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))}, \quad \text{i=0, 1, } \ldots, \\
\text{and} \quad \theta_n(t) &= \theta_n(\frac{i}{n}), \quad \text{if } \tfrac{i}{n} \leq t < \tfrac{i+1}{n} \text{ for some integer } i. \qquad (2.2)
\end{aligned}$$

Where $p_n(\frac{i}{n}) \approx 1 - \frac{p}{\sqrt{n}}$ for some $p > 0$.
The discrete time Standard MCMC will be embedded in a continuous time where the nth approximation is given by:

$$\begin{aligned}
X_n(0) &= x_0 \in \mathbf{R}; \\
X_n\Big(\frac{i+1}{n}\Big) &= X_n\Big(\frac{i}{n}\Big) + \frac{1}{\sqrt{n}}\theta_0\xi_n\Big(\frac{i+1}{n}\Big)\epsilon_n\Big(\frac{i+1}{n}\Big), \quad \text{i=0, 1, } \ldots, \ \theta_0 \in \mathbf{R}^+, \\
X_n(t) &= X_n\Big(\frac{i}{n}\Big), \quad \text{if } \tfrac{i}{n} \leq t < \tfrac{i+1}{n} \text{ for some integer } i. \qquad (2.3)
\end{aligned}$$

where $\xi_n\left(\frac{i}{n}\right)$ has the same conditional distribution. The theorem in the next section deals with the diffusion approximation of the above AMCMC and standard MCMC.

3 Main Theorem

THEOREM 1 *The limit of the process* $\mathbf{Y}_n(t) := \Big(X_n(t) \quad \theta_n(t)\Big)$, *where* $X_n(t)$ *and* $\theta_n(t)$ *is given by 2.1 and 2.2 respectively, is governed by the SDE:*

$$d\mathbf{Y}_t = b(\mathbf{Y}_t)dt + \sigma(\mathbf{Y}_t)d\mathbf{W}_t. \qquad (3.1)$$

Where,

$$b(\mathbf{Y}_t) = \left(\frac{\theta_t^2}{2}\frac{\psi'(X_t)}{\psi(X_t)}, \quad \theta_t\left(p - \frac{\theta_t}{\sqrt{2\pi}}\frac{|\psi'(X_t)|}{\psi(X_t)}\right)\right)^T,$$

and

$$\sigma(\mathbf{Y_t}) = \begin{pmatrix} \theta_t & 0 \\ 0 & 0 \end{pmatrix}$$

and $\mathbf{W}_t$ *is a two dimensional Wiener process.*

Proof.
Firstly, note that since $\mathbf{Y}_n(\frac{i}{n}) := \left(X_n(\frac{i}{n}) \ \ \theta_n(\frac{i}{n})\right)$ is a homogenous Markov chain it defines a transition kernel

$$\Pi_n(\mathbf{y}, A) = P\left(Y_n(\frac{i+1}{n}) \in A|\mathbf{Y}_n(\frac{i}{n}) = \mathbf{y}\right) \quad \forall \mathbf{y} \in \mathbf{R} \times \mathbf{R}^+ \text{ and } \forall A \in \mathbf{B}(\mathbf{R} \times \mathbf{R}^+)$$

The proof then follows essentially by obtaining the 'drift' and 'diffusion' coefficients of the discretised process and then finding its limit. Formally, first obtain the quantities :

$$\begin{aligned}
\mathbf{a}_n(\mathbf{y}, t) &:= (a_{n,i,j}(\mathbf{y}, t))_{i,j=1,2} := n \int_{\mathbf{R}} (\mathbf{z} - \mathbf{y})(\mathbf{z} - \mathbf{y})' \Pi_n(\mathbf{y}, d\mathbf{z}), \\
\mathbf{b}_n(\mathbf{y}, t) &:= (b_{n,k}(\mathbf{y}, t))_{k=1,2} := n \int_{\mathbf{R}} (\mathbf{z} - \mathbf{y}) \Pi_n(\mathbf{y}, d\mathbf{z}).
\end{aligned}$$

Then find matrix $\mathbf{a}$ and vector $\mathbf{b}$ such that $\lim_{n\to\infty} ||\mathbf{a}_n(\mathbf{y}, t) - \mathbf{a}(\mathbf{y}, t)|| = 0$ and $\lim_{n\to\infty} ||\mathbf{b}_n(\mathbf{y}, t) - \mathbf{b}(\mathbf{y}, t)||$ (where the norms are the usual matrix norm and vector norms respectively). Obtain the square root of matrix $\mathbf{a}(\mathbf{y}, t)$(say $\sigma_(\mathbf{y}, t)$), which satisfies $\mathbf{a}(\mathbf{y}, t) = \sigma(\mathbf{y}, t)\sigma(\mathbf{y}, t)'$. If these coefficients define an SDE uniquely which is non-explosive, then the limiting process is governed by the equation:

$$d\mathbf{Y}_t = \mathbf{b}(\mathbf{Y}_t, t)dt + \sigma(\mathbf{Y}_t, t)d\mathbf{W}_t.$$

where $\mathbf{W}_t$ is a two dimensional Wiener process. For the processes defined in 2.1 and 2.2 the quantities $\mathbf{a}_n(\mathbf{y}, t)$ and $\mathbf{b}_n(\mathbf{y}, t)$ is (for $\mathbf{y} = (x \ \ \theta)$):

$$\begin{aligned}
\lim_{n\to\infty} b_{n,1}(\mathbf{y}, t) &= \frac{\theta^2}{2} \frac{\psi'(x)}{\psi(x)}, \\
\lim_{n\to\infty} b_{n,2}(\mathbf{y}, t) &= \theta(p - \frac{\theta}{\sqrt{2\pi}} \frac{|\psi'(x)|}{\psi(x)}), \\
\lim_{n\to\infty} a_{n,1,1}(\mathbf{y}, t) &= \theta^2, \\
\lim_{n\to\infty} a_{n,2,2}(\mathbf{y}, t) &= 0, \\
\lim_{n\to\infty} a_{n,2,1}(\mathbf{y}, t) &= 0 = \lim_{n\to\infty} a_{n,1,2}(\mathbf{y}, t)
\end{aligned}$$

See Section 7.1 for the derivations.
Therefore $||\mathbf{a}_n(\mathbf{y}, t) - \mathbf{a}(\mathbf{y}, t)|| \to 0$ and $||\mathbf{b}_n(\mathbf{y}, t) - \mathbf{b}(\mathbf{y}, t)|| \to 0$ where

$$\begin{aligned}
\mathbf{a}(\mathbf{y}, t) &= \begin{pmatrix} \theta^2 & 0 \\ 0 & 0 \end{pmatrix} \Rightarrow \sigma(\mathbf{y}, t) = \begin{pmatrix} \theta & 0 \\ 0 & 0 \end{pmatrix} \\
\text{and } \ \mathbf{b}(\mathbf{y}, t) &= \left(\frac{\theta^2}{2} \frac{\psi'(x)}{\psi(x)}, \quad \theta(p - \frac{\theta}{\sqrt{2\pi}} \frac{|\psi'(x)|}{\psi(x)}) \right)^T.
\end{aligned}$$

This proves the theorem. ∎

Remark **2** *Therefore, following the same proof as that of Theorem 1 the SDE of the process corresponding to standard MCMC becomes*

$$dX_t = -\frac{\psi'(X_t)}{\psi(X_t)}\frac{\theta^2}{2}dt + \theta dW_t \tag{3.2}$$

In the next section we give extensive simulation for discrete and its diffusion approximation comparing adaptive and non-adaptive methods.

4 Comparison between Adaptive and Non-Adaptive MCMC by simulations

In this section we compare both the discrete and continuous non adaptive MH sampler against its adaptive counterpart as proposed in Section 2. We try to simulate samples from target distributions with heavier and lighter tails compared to the Normal distribution. In the standard MH the tuning parameter is kept fixed. Since the tuning parameter that gives the best result is not known in general, so we compare the simulations for a host of θ values. (Note that results relating to optimal p for Normal distribution is given in [7] which have been extended further in [3]).

4.1 Comparison between the discrete time chains

We generate the discrete time version of the adaptive and non adaptive sampler for different values of p and the starting θ. All the proposal distributions are normal and the target density is Normal(0,1) in Table 1, Cauchy(0,1) in Table 2 and $t(2)$ in Table 3. After generating a sample of size 10000 we discard the first 1000 samples as burn-in. To check the efficiency of the sampler we perform the one sample Kolmogorov-Smirnov (KS) test on the remaining sample and find the asymptotic p value of the KS test statistic D measuring the distance between the empirical distribution of the generated sample and the target distribution.

Another measure of the amount of mixing is **Expected Square Jumping Distance**(ESJD) defined as $E(X_i - X_{i-1})^2$. Based on the generated sample it can be estimated by $\frac{1}{n-B}\sum_{i=B+1}^{n}(X_i - X_{i-1})^2$, where B is the size of burn-in sample. In general, higher value of ESJD implies greater mixing.(see [8] for more details).

See Tables 1, 2, 3 here.

From Table (1) we see that the starting value of $\theta_0 = 2.38$ is the best choice with respect to the Non-Adaptive chain as well as the Adaptive chain. This value of θ was suggested by Gelman et al ([7]). For adaptive chain the optimal value of p lies somewhere between 0.25 and 0.50. Again, by [7], the optimal value of acceptance probability was close to 0.238. Therefore our simulations corroborates their findings to some extent.
It is well known that the naive MH algorithm is not efficient enough in simulating from a heavy tailed distribution. This is what we see in Table (2) and Table (3). But we see that for the adaptive version with p lying in the same interval, ie in [0.25,0.50], its performance is at least better than that of its non adaptive counterpart, although by itself it is not quite efficient.
In comparison we see that the continuous time version of the AMCMC performs much better than its discrete counterpart as elucidated in the next section.

4.2 Comparison between the continuous time processes

To comapre how fast the two SDEs 3.1 and 3.2 converge to stationarity we apply Euler discretisation to each of them for various choices of target density ψ and mesh size h. For the process 3.1 the Euler discretization is given by $X_{ih},\ \ i = 0, 1, 2 \ldots, \frac{T}{h}$ where:

$$\begin{aligned} X_{(i+1)h} &= X_{ih} + h\frac{\psi'(X_{ih})}{\psi(X_{ih})}\frac{\theta_{ih}^2}{2} + \sqrt{h}\theta_{ih}Z_{ih}^{(1)} \\ \theta_{(i+1)h} &= \theta_{ih} + h\Big(\theta_{ih}(p - \theta_{ih}\frac{|X_{ih}|}{\sqrt{2\pi}})\Big), \quad i = 0, 1, \ldots, T/h. \end{aligned}$$

Similarly the Euler discretisation for the process 3.2 is $\{Y_{ih}\}$ where:

$$Y_{(i+1)h} = Y_{ih} + h\frac{\psi'(X_{ih})}{\psi(X_{ih})}\frac{\theta^2}{2} + \sqrt{h}\theta Z_{ih}^{(2)}, \quad i = 0, 1, \ldots, T/h, \ \theta = \theta_0 \in \mathbf{R}^+$$

Where $Z_{ih}^{(j)},\ \ i = 0, 1, \ldots, T/h,\ \ j = 1, 2$ are independent $N(0, 1)$ random variables.

For various values of the mesh size h we simulate 1000 parallel SDE using the Euler discretisation for the Adaptive and the Standard MCMC and obtain the value of X_T at time $T = 1$. Table 4, Table (5) and Table 6 gives the result when the target

distribution is $N(0,1)$, $C(0,1)$ and $Exp(1)$ respectively. We also compute the Kolmogorov Smirnov distance between the sample and the target distribution and also find its asymptotic p value.

See Table 4, 5, 6 here.

It should be noted that since the support of the Exponential density (Table 5) is only the positive part of the real line, while simulating the SDE corresponding to the distribution any move to the left of zero was modified accordingly
The tables clearly indicate that for a proper choice of the parameter p, the Adaptive version performs better than the Non-Adaptive version. Almost always the asymptotic p-value is small for smaller values of p it reaches its peak at an optimum p and then again decreases.The reason is that if p is small then the quantity $-\theta_{ih}|X_{ih}|/\sqrt{2\pi}$ dominates p and θ_{ih} decreases on the average. On the other hand if p is large then p dominates and θ_{ih} increases in on the average. As a result the sample thus generated differs widely from the target density.
Another table of interest is Table 6. Standard MCMC is not quite adept in sampling from a density with heavy tails. AMCMC to some degree addresses this problem where we see that the p-value is always higher for the Adaptive case for all values of mesh size h.

5 Conclusion

Diffusion approximation is a well studied technique that has been applied in many fields (e.g., [16], [5]). In AMCMC the tuning parameter changes as the iteration progresses and therefore the transition kernel also changes. As a result the invariant properties of the chain are not easily obtainable, even if the transition kernel at each iteration is stationary. In this paper, we have applied the diffusion approximation procedure to the AMCMC chain and obtained the limiting SDEs to arrive at the target distribution. Diffusive limits for Metropolis Hastings algorithm were earlier obtained in [20, 21, 18]. Also, there are some recent work on diffusive limits of high-dimensional non adaptive MCMC that came to the attention of the authors (for example, see [13]). However, to the best of our knowledge, application of diffusion approximation to Adaptive MCMC and subsequent comparison between AMCMC and Standard MCMC using their respect diffusive limits have not been done earlier.

Our technique expands the scope of comparison between AMCMC and Standard MCMC, as embedding in continuous time allows various discrete approximations through which one can compare them in finer details.
Some tasks awaits future research. First, the existence of the invariant distribution of the SDE has to be verified theoretically for different choices of the target distribution. Second, normal proposal in case of heavy tailed ψ or multi modal densities is not always very efficient. Algorithms must be designed to handle such cases. It should be investigated whether combining the ideas of adaptivity and heavy tailed proposal increases the efficiency. There are some work in this area (see [11]). Third, although diffusion approximation simplifies the problem it does not say anything about the time to convergence of the chain. An interesting problem would be to bound the time taken to reach stationarity. For more information relating to bounding times of MCMC see [12]. We hope to take up these issues in our future research.

6 Acknowledgement

We thank Peter Green, B. V. Rao, LPS V 2010 workshop participants at ISI Bangalore, and seminar audience at the Department of Mathematics, University of Bristol for valuable comments.
We also thank an anonymous referee for his/her helpful suggestions and comments for the improvement of the paper in present form.

References

[1] Andrieu C. and Mouline E., (2006), On the ergodicity property of Adaptive MCMC, Annals of Applied Probability, 2006, **16**, 2, 1462-1505.

[2] Atchade Y. and Rosenthal J. On Adaptive Markov chain Monte carlo algorithm, Bernoulli **11(5)**, 2005, 815-828.

[3] Bedard M.,(2006) *On the robustness of optimal scaling in Metropois Hastings algorithm*, Ph. D dissertation, University of Toronto.

[4] Besag J., Green P., Higdon D., Mengersen K.,(1995), *Bayesian Computation and Stochastic Systems*, Statistical Science, **10**, 1, 3-66.

[5] Ethier S. N. and Kurtz T. *Markov Process: Characterisation and convergence*, Wiley Series in Probability and Mathematical statistics.

[6] Gelfand A. E. and Sahu S. K., (1994), *On Markov chain Monte Carlo acceleration*, Journal of Computational and Graphical Statistics, **3**, 261-276.

[7] Gelman A. Roberts G. O. and Wilks D (1997), *Weak convergence and Optimal Scaling of Random Walk Metropolis algorithm*, Annals of Applied Probbaility, **7**, 110-120.

[8] Gelamn A. and Pasarica C. (2010), *Adaptively scaling the Metropolis Algorithm using Expected Square Jumping Distance*, Statistica Sinica, **20, 1**, 343-364.

[9] Haario H., Saksman E. and Tamminen J. (2001), *An Adaptive Metropolis sampler*, Bernoulli **7, 2**, 223-242.

[10] Hastings W. K. (1970), *Monte Carlo sampling methods using Markov chains and their applications*, Biometrika, **57**, 97-109.

[11] Jarner S. F. and Roberts G. O. (2007), *Convergence of Heavy-tailed Monte Carlo Markov Chain Algorithms*, Scand. J of Statistics, **34, 4**, 781-815.

[12] Joulin A. and Ollivier Y. (2010), *Curvature, Concentration and Error estimates for Markov chain Monte Carlo*,The Annals of Probability, **38, 6**, 2418-2442.

[13] Mattingly J. C., Pillai N., Stuart A. M., *Diffusive Limits of Random Walk Metropolis Algorithm in High Dimension*, arXiv: 1003.43006v3[math.PR].

[14] Metropolis N., Rosenbluth A. , Rosenbluth M., Teller A., and Teller E (1953), *Equations of state calculations by fast computing machines*, J.Chem. Phys. 21, pp. 1087-1091.

[15] Monro S. Robbins (1951), *A Stochastic Approximation Algorithm*, The Annals of Mathematical Statistics, **22, 3**, 400-407.

[16] Nelson D. B.(1990), *ARCH models as diffusion approximation* , Journal of Econometrics, **45,1-2**, 7-38.

[17] Roberts G. and Rosenthal J.(2007), *Coupling and Ergodicity in Adaptive MCMC*, Journal of Applied Probability **44**, 458-475.

[18] Roberts G. O. and Tweedie R. L. (1996), *Exponential convergence of Langevin diffusions and their discrete approximations*, Bernoulli, **2**, 341-364.

[19] Rosenthal J. and Roberts G.O (2009), *Examples of Adaptive MCMC*, Journal of Computational and Graphical Statistics, 2009, 18(2), 349-367. doi:10.1198/jcgs.2009.06134.

[20] Stramer O. and Tweedie R. L., (1999a), *Langevin type models I: Diffusion with given stationary distribution and their discretisation*, Methodology and Computing in Applied Probability, 1:3, 283-306.

[21] Stramer O. and Tweedie R. L., (1999b), *Langevin type models II: Self targeting candidates for MCMC algorithms*, Methodology and Computing in Applied Probability, 1:3, 307-328.

[22] Tierney L. (1994), Markov chains for exploring posterior distributions (with discussion), Ann. Stat., 22, 1701-1762.

7 Appendix

7.1 Drift and diffusion coefficients

Writing $\mathbf{y} = (x \quad \theta)$ we have

7.1.1 $b_{n,1}$

$$\begin{aligned}
b_{n,1}(\mathbf{y},t) &= nE(X_n(\frac{i+1}{n}) - X_n(\frac{i}{n})|\ \mathbf{Y}_n(\frac{i}{n}) = \mathbf{y}),\ \ \forall i = 0,1,\ldots,\forall n \geq 1 \\
&= E(\sqrt{n}\theta_n(\frac{i}{n})\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})|\ \mathbf{Y}_n(\frac{i}{n}) = \mathbf{y}) \\
&= \sqrt{n}\theta E(\xi_n(\frac{i}{n})\epsilon_n(\frac{i}{n})|\ \mathbf{Y}_n(\frac{i}{n}) = \mathbf{y}) \\
&= \sqrt{n}\theta\Big(E(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})I_A|\ X_n(\frac{i}{n}) = x,\ \theta_n(\frac{i}{n}) = \theta) \\
&+ E(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})I_{A^c}|\ X_n(\frac{i}{n}) = x,\ \theta_n(\frac{i}{n}) = \theta)\Big).
\end{aligned}$$

where $A_n(= A_n(x,\theta))$ is the set where $\xi_n(\frac{i+1}{n})$ is one with probability 1, i.e,

$$\begin{aligned}
A_n(x,\theta) &= \{y : \frac{\psi(x + \frac{1}{\sqrt{n}}\theta y)}{\psi(x)} > 1\}. \\
\text{And} \lim_{n\to\infty} A_n(x,\theta) &= \begin{cases} (-\infty, 0) & \text{if } \psi'(x) > 0 \\ (0,\infty) & \text{if } \psi'(x) < 0 \end{cases}.
\end{aligned}$$

Therefore,

$$
\begin{aligned}
b_{n,1}(\mathbf{y},t) &= \sqrt{n}\theta\Big(\int_{A_n}\epsilon\phi(\epsilon)d\epsilon + \int_{A_n^c}\frac{\psi(x+\frac{1}{\sqrt{n}}\theta\epsilon)}{\psi(x)}\epsilon\phi(\epsilon)d\epsilon\Big)\\
&= \sqrt{n}\theta\Big(\int_{A_n}\epsilon\phi(\epsilon)d\epsilon + \int_{A_n^c}\epsilon\phi(\epsilon)d\epsilon + \frac{\theta}{\sqrt{n}}\frac{\psi'(x)}{\psi(x)}\int_{A_n^c}\epsilon^2\phi(\epsilon)d\epsilon\\
&+ \frac{\theta^2}{n}\frac{\psi''(x)}{\psi(x)}\int_{A_n^c}\epsilon^3\phi(\epsilon)d\epsilon + O(\frac{1}{n})\Big)\ \text{(By Taylor's expansion)}\\
&= \sqrt{n}\theta\Big(\int_{\mathbf{R}}\epsilon d\epsilon + \frac{\theta}{\sqrt{n}}\frac{\psi'(x)}{\psi(x)}\int_{A_n^c}\epsilon^2\phi(\epsilon)d\epsilon + O(\frac{1}{n})\Big)\\
&= \theta^2\frac{\psi'(x)}{\psi(x)}\int_{A_n^c}\epsilon^2\phi(\epsilon)d\epsilon + O(\frac{1}{\sqrt{n}})\\
\Rightarrow \lim_{n\to\infty} b_{n,1}(\mathbf{y},t) &= \theta^2\frac{\psi'(x)}{\psi(x)}\lim_{n\to\infty}\int_{A_n^c}\epsilon^2 d\epsilon = \begin{cases}\theta^2\frac{\psi'(x)}{\psi(x)}\int_{-\infty}^0\epsilon^2\phi(\epsilon)d\epsilon & \text{if } \psi'(x)>0\\ \theta^2\frac{\psi'(x)}{\psi(x)}\int_0^\infty\epsilon^2\phi(\epsilon)d\epsilon & \text{if } \psi'(x)<0\end{cases}\\
&= \frac{\theta^2}{2}\frac{\psi'(x)}{\psi(x)}.
\end{aligned}
$$

7.1.2 $b_{n,2}$

$$
\begin{aligned}
b_{n,2}(\mathbf{y},t) &= nE(\theta_n(\frac{i+1}{n}) - \theta_n(\frac{i}{n})|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y})\quad \forall i=0,1,\ldots\\
&= nE\Big(\theta_n(\frac{i}{n})\{e^{\frac{1}{\sqrt{n}}(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))}-1\}|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\\
&= n\theta\Big(\frac{1}{\sqrt{n}}E(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n})|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y})\\
&+ E(\frac{1}{2n}(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y})\Big)\\
&= \theta\sqrt{n}E(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n})|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y})\\
&+ \frac{\theta}{2}E((\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))^2)|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y})\\
&+ O(\frac{1}{\sqrt{n}})..
\end{aligned}
$$

Now,

$$
\begin{aligned}
&\theta\sqrt{n}E(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n})|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y})\\
=&\ \theta\sqrt{n}E(\xi_n(\frac{i+1}{n})|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y})-p_n(\frac{i}{n})\\
=&\ \theta\sqrt{n}\Big(\int_{A_n}\phi(\epsilon)d\epsilon+\int_{A_n^c}\frac{\psi(x+\frac{1}{\sqrt{n}}\theta\epsilon)}{\psi(x)}\phi(\epsilon)d\epsilon-p_n(\frac{i}{n})\Big)
\end{aligned}
$$

$$
\begin{aligned}
&= \theta\sqrt{n}\Big(\int_{A_n}\phi(\epsilon)d\epsilon + \int_{A_n^c}\{1+\frac{\theta}{\sqrt{n}}\frac{\psi'(x)}{\psi(x)}\epsilon + O(\frac{1}{n})\}\phi(\epsilon)d\epsilon - p_n(\frac{i}{n})\Big)\\
&= \theta\sqrt{n}(1-p_n(\frac{i}{n})) + \theta^2\frac{\psi'(x)}{\psi(x)}\int_{A_n^c}\epsilon\phi(\epsilon)d\epsilon + O(\frac{1}{\sqrt{n}}). \qquad (7.1)
\end{aligned}
$$

$$
\begin{aligned}
& E\Big((\xi_n(\frac{i+1}{n}) - p_n(\frac{i}{n}))^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\\
=\ & E\Big(\xi_n(\frac{i+1}{n})^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big) - 2p_n(\frac{i}{n})E\Big(\xi_n(\frac{i+1}{n})|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big) + p_n(\frac{i}{n})^2\\
=\ & \int_{A_n}\phi(\epsilon)d\epsilon + \int_{A_n^c}\frac{\psi(x+\frac{1}{\sqrt{n}}\theta\epsilon)}{\psi(x)}\phi(\epsilon)d\epsilon - 2p_n(\frac{i}{n})\Big(\int_{A_n}\phi(\epsilon)d\epsilon + \int_{A_n^c}\frac{\psi(x+\frac{1}{\sqrt{n}}\theta\epsilon)}{\psi(x)}\phi(\epsilon)d\epsilon\Big)\\
+\ & p_n(\frac{i}{n})^2\\
=\ & (1-p_n(\frac{i}{n}))^2 + \frac{1}{\sqrt{n}}(1-2p_n(\frac{i}{n}))\theta\frac{\psi'(x)}{\psi(x)}\int_{A_n^c}\epsilon\phi(\epsilon)d\epsilon + O(\frac{1}{n}). \qquad (7.2)
\end{aligned}
$$

Adding 7.1 and 7.2 we have,

$$
\begin{aligned}
b_{n,2} &= \theta\sqrt{n}(1-p_n(\frac{i}{n})) + (1-p_n(\frac{i}{n}))^2 + \Big(\theta^2\frac{\psi'(x)}{\psi(x)} + \frac{1}{\sqrt{n}}(1-2p_n(\frac{i}{n}))\theta\frac{\psi'(x)}{\psi(x)}\Big)\int_{A_n^c}\epsilon\phi(\epsilon)d\epsilon.\\
&+ O(\frac{1}{n})
\end{aligned}
$$

.

Recalling that $1-p_n(\frac{i}{n}) \approx \frac{p}{\sqrt{n}} \Rightarrow \frac{1}{\sqrt{n}}(1-2p_n(\frac{i}{n})) \approx \frac{1}{\sqrt{n}}(2p-1)$ we have,

$$
\begin{aligned}
\lim_{n\to\infty}\mathbf{b}_{n,2}(\mathbf{y},t) &= \theta p + \theta^2\frac{\psi'(x)}{\psi(x)}\lim_{n\to\infty}\int_{A_n^c}\epsilon\phi(\epsilon)d\epsilon\\
&= \begin{cases} \theta\Big(p+\frac{\theta}{\sqrt{2\pi}}\frac{\psi'(x)}{\psi(x)}\Big) & \text{if } \psi'(x)<0\\ \theta\Big(p-\frac{\theta}{\sqrt{2\pi}}\frac{\psi'(x)}{\psi(x)}\Big) & \text{if } \psi(x)>0\end{cases}\\
&= \theta\Big(p - sgn(\psi'(x)\frac{\theta}{\sqrt{2\pi}}\frac{\psi'(x)}{\psi(x)}\Big)\\
&= \theta\Big(p - \frac{\theta}{\sqrt{2\pi}}\frac{|\psi'(x)|}{\psi(x)}\Big)
\end{aligned}
$$

.

7.1.3 $a_{n,1,1}$

$$
\begin{aligned}
a_{n,1,1}(\mathbf{y},t) &= nE\Big((X_n(\frac{i+1}{n}) - X_n(\frac{i}{n})^2)|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\ \forall i=0,1,\ldots\\
&= nE\Big(\frac{\theta^2}{n}\xi_n(\frac{i+1}{n})^2\epsilon_n(\frac{i+1}{n})^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)
\end{aligned}
$$

$$
\begin{aligned}
&= \theta^2 E(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y})\\
&= \theta^2\Big(E(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})^2 I_{A_n}|\ \mathbf{Y}_n(\frac{i}{n})=\mathbf{y})\\
&+\ E(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})^2 I_{A_n^c}|\ \mathbf{Y}_n(\frac{i}{n})=\mathbf{y})\Big)\\
&= \theta^2\Big(\int_{A_n}\epsilon^2 d\epsilon+\int_{A_n^c}\frac{\psi(x+\frac{1}{\sqrt{n}}\theta\epsilon)}{\psi(x)}d\epsilon\Big)\\
&= \theta^2\Big(\int_{A_n}\epsilon^2 d\epsilon+\int_{A_n^c}\epsilon^2 d\epsilon+O(\frac{1}{\sqrt{n}})\Big)\\
&= \theta^2+O(\frac{1}{\sqrt{n}}).\\
\Rightarrow \lim_{n\to\infty} a_{n,1,1}(t) &= \theta^2..
\end{aligned}
$$

7.1.4 $a_{n,2,2}$

$$
\begin{aligned}
a_{n,2,2}(\mathbf{y},t) &= nE\Big((\theta_n(\frac{i+1}{n})-\theta_n(\frac{i}{n}))^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\\
&= nE\Big(\theta_n(\frac{i}{n})^2(e^{\frac{1}{\sqrt{n}}(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))}-1)^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\\
&= n\theta^2 E\Big(\{\frac{1}{\sqrt{n}}(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))+\frac{1}{2n}(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))^2\}^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\\
&= \theta^2 E\Big((\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)+o(n)\\
\Rightarrow \lim_{n\to\infty} a_{n,2,2}(\mathbf{y},t) &= \theta^2\lim_{n\to\infty}E\Big((\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))^2|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\\
&= 0 \ \text{ from 7.2.}
\end{aligned}
$$

7.1.5 $a_{n,1,2}$

$$
\begin{aligned}
a_{n,1,2}(\mathbf{y},t) &= nE\Big(\{X_n(\frac{i+1}{n})-X_n(\frac{i}{n})\}\{\theta_n(\frac{i+1}{n})-\theta_n(\frac{i}{n})\}|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\\
&= nE\Big(\{\frac{1}{\sqrt{n}}\theta_n(\frac{i}{n})\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})\}\{\theta_n(\frac{i}{n})(e^{\frac{1}{\sqrt{n}}(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))})\}\Big)\\
&= \sqrt{n}\theta^2 E\Big(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})\{\frac{1}{\sqrt{n}}(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))\\
&+ \frac{1}{2n}(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))^2+\}|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\\
&= \theta^2 E\Big(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})(\xi_n(\frac{i+1}{n})-p_n(\frac{i}{n}))|\mathbf{Y}_n(\frac{i}{n})=\mathbf{y}\Big)\\
&+ O(\frac{1}{\sqrt{n}}) \ \text{ from 7.2.}
\end{aligned}
$$

$$
\begin{aligned}
\text{Now,} \quad & E\Big(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})(\xi_n(\frac{i+1}{n}) - p_n(\frac{1}{n}))|\mathbf{Y}_n(\frac{i}{n}) = \mathbf{y}\Big) \\
= \quad & E\Big(\xi_n(\frac{i+1}{n})^2\epsilon_n(\frac{i+1}{n})|\mathbf{Y}_n(\frac{i}{n}) = \mathbf{y}\Big) \\
- \quad & p_n(\frac{1}{n})E\Big(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})|\mathbf{Y}_n(\frac{i}{n}) = \mathbf{y}\Big) \\
= \quad & (1 - p_n(\frac{1}{n}))E\Big(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})|\mathbf{Y}_n(\frac{i}{n}) = \mathbf{y}\Big) \\
= \quad & (1 - p_n(\frac{1}{n}))E\Big(\xi_n(\frac{i+1}{n})\epsilon_n(\frac{i+1}{n})|\mathbf{Y}_n(\frac{i}{n}) = \mathbf{y}\Big) \\
= \quad & (1 - p_n(\frac{1}{n}))\Big(\int_{A_n} \epsilon\phi(\epsilon)d\epsilon + \int_{A_n^c} \frac{\psi(x + \frac{1}{\sqrt{n}}\theta\epsilon)}{\psi(x)}\epsilon\phi(\epsilon)d\epsilon\Big) \\
= \quad & (1 - p_n(\frac{1}{n}))\Big(\int_{A_n} \epsilon\phi(\epsilon)d\epsilon + \int_{A_n^c} \epsilon\phi(\epsilon)d\epsilon + \frac{1}{\sqrt{n}}\frac{\psi'(x)}{\psi(x)}\theta\int_{A_n^c} \epsilon^2\phi(\epsilon)d\epsilon \\
+ \quad & O(\frac{1}{n})\Big) \\
= \quad & \frac{1}{\sqrt{n}}(1 - p_n(\frac{1}{n})))\frac{\psi'(x)}{\psi(x)}\theta\int_{A_n^c} \epsilon^2\phi(\epsilon)d\epsilon + O(\frac{1}{n}) \\
\Rightarrow \lim_{n\to\infty} a_{n,1,2} = a_{n,2,1} \quad = \quad & 0.
\end{aligned}
$$

8 Tables

θ	p	Adpative D	p-value	E	Non-Adaptive D	p-value	E
	0.10	0.0502	2.2e-12	0.2388			
	0.25	0.0278	1.729e-6	0.6064			
0.10	0.50	0.0165	0.01472	0.7150	0.1369	2.2e-16	0.00934
	0.75	0.025	2.702e-5	0.3709			
	0.10	0.0391	2.244e-12	0.2587			
	0.25	0.0258	1.199e-5	0.5926			
0.25	0.50	0.024	6.507e-5	0.6971	0.0377	1.56e-11	0.05228
	0.75	0.0273	3.006e-6	0.3618			
	0.10	0.0458	2.2e-16	0.2356			
	0.25	0.0186	0.003867	0.5835			
1.0	0.50	0.0168	0.01270	0.7137	0.0251	2.29e-5	0.46209
	0.75	0.0272	3.344e-6	0.3642			
	0.10	0.039	2.46e-16	0.2572			
	0.25	0.0138	0.06627	0.6050			
2.38	0.50	0.0153	0.02883	0.7070	0.0223	2.69e-4	0.71047
	0.75	0.0233	0.00010	0.3648			
	0.10	0.0467	1.066e-14	0.0427			
	0.25	0.0374	2.38e-11	0.5953			
10	0.50	0.0221	3.012e-4	0.6988	0.0302	1.444e-7	0.26411
	0.75	0.0208	0.00081	0.3613			
	0.10	0.0467	2.2e-16	0.2433			
	0.25	0.0272	3.42e-16	0.5894			
20	0.50	0.030	1.361e-7	0.6992	0.0669	2.26e-15	0.14603
	0.75	0.0225	0.002141	0.3709			

Table 1: Table comparing the asymptotic p values of sample generated using AMCMC and MCMC for different values of p and θ where the target density is Normal(0,1)

θ	p	Adpative D	p-value	E	Non-Adaptive D	p-value	E
	0.10	0.0675	2.2e-16	34.8595			
	0.234	0.0246	3.65e-5	10.6793			
0.10	0.50	0.0272	3.419e-6	2.9033	0.2023	2.2e-16	0.00942
	0.75	0.0335	3.299e-9	0.7969			
	0.10	0.0582	2.2e-16	81.86962			
	0.234	0.0389	2.79e-12	10.3054			
0.25	0.50	0.0186	0.00399	2.9030	0.1069	2.2e-16	0.5468
	0.75	0.0385	5.29e-12	0.7305			
	0.10	0.0455	2.2e-16	34.8598			
	0.234	0.0332	4.601e-9	10.7424			
1.0	0.50	0.0227	1.938e-4	2.9033	0.0418	4.47e-14	0.6453
	0.75	0.0336	2.81e-9	0.7144			
	0.10	0.0675	2.2e-16	34.8598			
	0.234	0.024	6.175e-5	11.0824			
2.38	0.50	0.0213	5.7137e-4	2.7663	0.0302	1.493e-7	1.01557
	0.75	0.0353	3.646e-10	0.7490			
	0.10	0.0528	2.2e-16	34.8598			
	0.234	0.0383	6.56e-12	10.0612			
10	0.50	0.0192	0.00258	2.8593	0.0334	4.028e-9	10.19617
	0.75	0.0318	2.538e-8	0.7582			
	0.10	0.0529	2.2e-16	19.5438			
	0.234	0.034	1.836e-9	10.4076			
20	0.50	0.0224	3.831e-5	2.8915	0.0415	6.40e-14	22.9938
	0.75	0.0306	9.313e-8	0.7419			

Table 2: Table comparing the asymptotic p values of sample generated using AMCMC and MCMC for different values of p and θ where the target density is Cauchy(0,1)

θ	p	Adpative D	p-value	E	Non-Adaptive D	p-value	E
0.10	0.10	0.0605	2.2e-16	26.85993	0.1942	2.2e-16	0.0094
	0.234	0.0506	2.2e-16	10.28817			
	0.50	0.0563	2.2e-16	2.8169			
	0.75	0.0488	2.2e-16	0.7969			
0.25	0.10	0.076	2.2e-16	81.86962	0.1365	2.2e-16	0.0546
	0.234	0.0717	2.2e-16	10.3054			
	0.50	0.0504	2.2e-16	2.9030			
	0.75	0.0436	2.554e-15	0.7305			
1.0	0.10	0.0788	2.2e-16	23.0392	0.0921	2.2e-16	0.6253
	0.234	0.0641	2.2e-16	10.0612034			
	0.50	0.0528	2.2e-16	2.9537			
	0.75	0.0645	2.2e-16	0.7529			
2.38	0.10	0.0675	2.2e-16	24.32133	0.0652	2.2e-16	1.9155
	0.234	0.024	2.2e-16	10.61035			
	0.50	0.0559	2.2e-16	2.8804			
	0.75	0.0398	8.202e-13	0.6986			
10	0.10	0.0498	2.2e-16	24.6033	0.0566	2.2e-16	10.1961
	0.238	0.0576	2.2e-16	10.5307			
	0.50	0.0489	2.2e-16	2.7374			
	0.75	0.0483	2.2e-16	0.7206			
20	0.10	0.0549	2.2e-16	24.7967	0.062	2.2e-16	22.9938
	0.234	0.0565	2.2e-16	11.0529			
	0.50	0.0557	2.2e-16	2.7514			
	0.75	0.0511	2.2e-16	0.7322			

Table 3: Table comparing the asymptotic p values of sample generated using AMCMC and MCMC for different values of p and θ where the target density is t(2)

p	p value		$\theta(T)$	D	
	Adaptive	Non-Adaptive		Adaptive	Non-Adaptive
h=0.0001					
1.0	0.0385		4.5900	0.0444	
2.0	**0.2035**	**0.5273**	**0.0338**	**0.0338**	**0.0256**
2.5	0.1415		8.8963	0.0364	
h=0.0005					
4.5	0.4681		12.8575	0.0268	
5.0	**0.4774**	**0.28**	**14.4240**	**0.0266**	**0.0313**
5.5	0.4186		15.9708	0.0279	
6.0	0.2254		17.50510	0.033	0.0313
h=0.001					
4	0.1999		15.7125	0.0305	
5	0.3804		14.0184	0.02870	
5.5	0.3409		20.9751	0.0297	
6.0	0.4179		17.6700	0.0279	
6.5	**0.4393**	**0.628**	**18.9813**	**0.0274**	**0.0237**
7.5	0.3369		21.6681	0.0298	
8.0	0.2127		23.0411	0.0335	
h=0.005					
0.2	0.2900		2.4322	0.0310	
0.5	0.2595		2.7888	0.0319	
1.0	0.1761		3.5346	0.0348	
1.5	**0.4378**	**6.08e-10**	**4.5760**	**0.0275**	**0.096**
2.0	0.3782		5.9303	0.0288	
2.5	0.2489		7.3848	0.0323	
3.0	0.08535		8.8085	0.0397	
h=0.01					
1.0	0.0033		3.6390	0.0565	
1.5	0.0180		5.0818	0.0485	
2.0	**0.0977**	**0**	**6.4826**	**0.0388**	**0.4516**.
3.0	0.0347		9.2409	0.0450	

Table 4: Simulation of the SDE for Normal target desnity

p	p value		$\theta(T)$	D	
	Adaptive	Non-Adaptive		Adaptive	Non-Adaptive
h=0.0001					
1.5	0.1885		4.5917	0.0344	
2.0	0.4301		5.5788	0.0276	
2.5	**0.4880**	**0.3641**	**6.6407**	**0.0264**	**0.0291**
3.0	0.3783		7.7608	0.0288	
20	0.0507		50.13257	0.0429	
h=0.0005					
2.0	0.02508		5.5782	0.0468	
2.5	0.07267		6.6402	0.0407	
3.0	0.1024		7.7604	0.0385	
3.5	0.2379		8.9240	0.0326	
4.0	**0.7415**	**0.6055**	**10.1186**	**0.0216**	**0.0241**
4.25	0.4527		10.7244	0.0271	
4.5	0.04414		11.3345	0.0274	
h=0.001					
1.5	0.6933		4.5905	0.0225	
2.0	**0.9175**	**0.3688**	**5.5774**	**0.0176**	**0.029**
2.5	0.5089		6.6394	0.026	
h=0.005					
5.5	0.328		13.80297	0.03	
6.0	**0.9579**	**0.893**	**15.0483**	**0.0161**	**0.0183**
6.5	0.4649		16.2972	0.0269	
h=0.01					
6.0	0.6368		15.0477	0.0235	
6.5	**0.9041**	**0.4622**	**16.2969**	**0.0179**	**0.0269**
7	0.05136	0.4622	17.548	0.0428	

Table 5: Simulation of the SDE for exponential target desnity

p	p value		$\theta(T)$	D	
	Adaptive	Non-Adaptive		Adaptive	Non-Adaptive
h=0.0001					
5	0.2938		18.13893	0.0309	
7	**0.4407**	**0.7756**	**25.15494**	**0.0274**	**0.0209**
8	0.3306		27.8692	0.03	
h=0.0005					
3.0	1.10e-11		14.5622	0.1139	
4.0	0.07		22.2738	0.0409	
4.5	0.1104		26.3661	0.0381	
6.0	**0.2259**	**0.4709**	**35.3313**	**0.033**	**0.0268**
7.0	0.2022		41.4020	0.0338	
h=0.001					
0.5	0.1423		4.2115	0.0363	
5	0.168		18.5107	0.0352	
6	**0.2353**	**0.4894**	**21.3778**	**0.0327**	**0.0264**
7	0.1641		24.8888	0.0354	
h=0.005					
2	0.0356		10.0997	0.0457	
2.5	0.0481		9.9744	0.0481	
3	0.0824		18.6841	0.0399	
3.5	**0.1197**	**1.652e-6**	**24.7255**	**0.0375**	**0.0837**
4.0	0.0627		31.845	0.0416	
h=0.01					
0.5	0.0762		3.8347	0.0404	
1.0	0.0162		4.8314	0.0491	
2.0	0.0277		8.1943	0.0462	
2.5	0.0060		9.9910	0.0539	
2.75	**0.0129**	**0**	**10.7327**	**0.0502**	**0.1537**
3.0	0.0079		11.2480	0.0526	
3.5	0.0004		12.9623	0.06491	

Table 6: Simulation of the SDE for Cauchy target desnity